3. The method of claim 2, further comprising:
when the transition ID is not in the cache,
receiving a plurality of pings from the first client at regular intervals, and
saving the document to the server a plurality of times without incurring reads/writes to a database, and
when the transition ID is in the cache,
receiving a plurality of pings from the first client to collect a lock table from a database to identify a second client, and
in a separate web service request, receiving the second client's lock information.

4. The method of claim 3, wherein receiving the second client's lock information comprises receiving a ping from the second client.

5. The method of claim 1, wherein switching from the single-client mode to the multi-client mode including a second client further comprises:
attempting to take another short-term lock on the document and seeing that the second client has already received the document;
determining if the multi-client mode is in progress by checking a database.

6. The method of claim 5, further comprising:
when the multi-client mode is not in progress,
sending the document and transition ID to the second client,
writing another transition ID to the transition table, and
writing to the database information indicating the second client has joined the editing session; and
when the multi-client mode is in progress,
sending the document and transition ID to other clients who are part of the editing session, and
writing to the database information indicating the second client has joined the editing session.

7. The method of claim 1 further comprising refreshing the cache by updating the transition table.

8. The method of claim 7, wherein updating the transition table comprises:
pinging the cache to check for the presence of transition ID in the cache; and
when the transition ID is not present in the cache, determining if the cache has been refreshed within a predetermined time interval;
when the cache has been refreshed within the predetermined time interval, respond to the ping from the cache, and
when the cache has not been refreshed within the predetermined time interval, fetch the transition table corresponding to the document's content database.

9. A system including a cache infrastructure for retrieval of presence metadata, the system comprising:
a memory storage unit; and
a processing unit coupled to the memory storage unit, wherein the processing unit is operative to:
receive short-term check out metadata from a first client to begin an editing session;
add a transition ID to the short-term check out metadata;
write the transition ID to a transition table stored in a cache,
switch from a single-client mode to a multi-client mode, wherein switching from the single-client mode to the multi-client mode comprises the processing unit being operative to notice when the first client attempts to take another short-term lock on the document and seeing that that a second client has already received the document;
determine if the multi-client mode is in progress by checking a database;
receive a ping from the first client to determine if the transition ID is in the cache;
when the transition ID is not in the cache,
receive pings from the first client at regular intervals, and
save the document to the server a plurality of times without incurring any reads/writes to the database, and
when the transition ID is in the cache,
receive a ping from the first client to collect a lock table from the database to identify the second client, and
receive the second client's lock information by receiving pings that do not carry lock information.

10. The system of claim 9, further comprising the processing unit being operative to:
when the multi-client mode is not in progress,
send the document and transition ID to the second client,
write the transition ID to the transition table, and
write to the database information indicating the second client has joined the editing session; and
when the multi-client mode is in progress,
send the document and transition ID to other clients that are part of the editing session, and
write to the database information indicating the second client has joined the editing session.

11. The system of claim 9, further comprising the processing unit being operative to refresh the cache by updating the transition table.

12. The system of claim 11, wherein updating the transition table comprises the processing unit being operative to:
receive a ping from the first client to check for the presence of transition ID in the cache; and
when the transition ID is not present in the cache, determine if the cache has been refreshed within a predetermined time interval;
when the cache has been refreshed within the predetermined time interval, respond to the ping from the cache, and
when the cache has not been refreshed within the predetermined time interval, fetch the transition table corresponding to the document's content database.

13. A client computer for retrieval of metadata relating to a multi-client editing session, the client computer comprising:
a memory unit; and
a processing unit operative to:
receive a document from a storage device, the document comprising short-term check out metadata indicating an editing session has begun;
ping a cache to determine if another transition ID is stored in the cache; and
send a transition ID to a transition table stored in a cache when the another transition ID is not stored in the cache to switch from a single-client mode to a multi-client mode when the another transition ID is stored in the cache.

14. The client computer of claim 13, wherein, when the transition ID is not in the cache the processing unit is further operative to: